UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/765,916	01/18/2001	Frederic Canut	CA7017922001	8270

55497   7590   02/27/2009
VISTA IP LAW GROUP LLP
1885 Lundy Avenue
Suite 108
SAN JOSE, CA 95131

EXAMINER
KANG, INSUN

ART UNIT	PAPER NUMBER
2193	

MAIL DATE	DELIVERY MODE
02/27/2009	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

_____

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

_____

Ex parte FREDERIC CANUT and MUSTAPHA DERRAS

_____

Appeal 2008-3796
Application 09/765,916
Technology Center 2100

_____

Decided:[1] February 27, 2009

_____

Before JAMES D. THOMAS, JOSEPH L. DIXON, and
ST. JOHN COURTENAY III, *Administrative Patent Judges*.

COURTENAY, Administrative Patent Judge.

DECISION ON APPEAL

---

[1] The two-month time period for filing an appeal or commencing a civil action, as recited in 37 CFR § 1.304, begins to run from the decided date shown on this page of the decision. The time period does not run from the Mail Date (paper delivery) or Notification Data (electronic delivery).

This is a decision on appeal under 35 U.S.C. § 134(a) from the Examiner's rejection of claims 1-26. We have jurisdiction under 35 U.S.C. § 6(b). We affirm.

STATEMENT OF THE CASE

THE INVENTION

Appellants' invention relates to the field of software development. More particularly, Appellants' invention relates to a system and methods for software code optimization. (Spec. 1).

Independent claim 1 is illustrative:

1.   A method of optimizing a software program for a target processor to meet performance objectives, where the software program is coded in a high-level language, the method comprising the steps of:

(a) optimizing the software program such that a resulting first optimized form of the software program is completely independent of the target processor and is at least partially coded in the high-level language, determining a first performance profile for the first optimized form of the software program, and comparing the first performance profile with the performance objectives;

(b) based on the results of comparing the first performance profile with the performance objectives, if the performance objectives are not met by the first optimized form of the software program, then optimizing the first optimized form of the software program such that a resulting second optimized form of the software program includes at least one portion that is dependent on the target processor and is coded in the high-level language; and

(c) flagging the at least one portion to indicate that the at least one portion is dependent on the target processor if the first optimized form of the software program is optimized to create the second optimized form of the software program.

PRIOR ART

Pieper   US 2003/0005419 A1   Jan. 2, 2003

Cain et al., "Portable Software Library Optimization," MITRE TECHNICAL REPORT, Feb., 1998. (hereinafter "Cain").

Kum et al., "A Floating-Point To Integer C Converter with Shift Reduction for Fixed-Point Digital Signal Processors," IEEE, 0-7803-5041-3/99, 1999. (hereinafter "Kum").

THE REJECTIONS

1. Claims 1-3, 8-16, and 21-26 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the combination of Pieper and Cain.

2. Claims 4-7 and 17-20 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over the combination of Pieper, Cain, and Kum.

GROUPING OF CLAIMS

In the principal Brief, Appellants present arguments for independent claim 1 as being representative of claims 2-3, 8-16, and 21-26 (App. Br. 6-7). Regarding the second obviousness rejection, Appellants submit that "claims 4-7 and 17-20 are patentable over Pieper in view of Cain and Kum under 35 U.S.C. §103 due to their respective dependence on present independent claims 1 and 14 . . . ." (App. Br. 7). We will, therefore, treat claims 1-26 as standing or falling with representative claim 1. *See* 37 C.F.R. § 41.37(c)(1)(vii).

APPELLANTS' CONTENTIONS

Regarding representative claim 1, Appellants contend that Pieper does not disclose or suggest the limitation of "a resulting first optimized form of the software program is completely independent of the target processor and is at least partially coded in the high-level language." (App. Br. 5). In particular, Appellants contend that Pieper's teaching of optimized code 60 (that is *substantially independent* of the architecture of the target processor) does not meet the language of claim 1 that requires a first optimized form of the software program that is *completely independent* of the target processor (App. Br 5; *see also* Pieper, para. [0030]).

EXAMINER'S RESPONSE

The Examiner avers that "[o]ne skilled in the pertinent art would know that a high-level language compiler forms intermediate code which is machine independent and optimizes the intermediate code in the middle end of the compilation stage, while formulating the target object code, which is also optimized by using machine-specific optimizations at a later stage of the compilation." (Ans. 11, ¶2, ll. 4-8).

ISSUE

Based upon our review of the administrative record, we have determined that the following issue is dispositive in this appeal:

Have Appellants shown the Examiner erred in finding that the combination of Pieper and Cain teaches and/or suggests the argued limitation of "a resulting first optimized form of the software

program is completely independent of the target processor?" (*See* independent claims 1 and 14).

PRINCIPLES OF LAW

"What matters is the objective reach of the claim. If the claim extends to what is obvious, it is invalid under *§ 103*." *KSR Int'l Co. v. Teleflex, Inc.*, 127 S. Ct. 1727, 1742 (2007). To be nonobvious, an improvement must be "more than the predictable use of prior art elements according to their established functions." *Id.* at 1740.

FINDINGS OF FACT

In our analysis *infra*, we rely on the following findings of fact (FF) that are supported by a preponderance of the evidence:

THE PRIMARY PIEPER REFERENCE

1. Pieper teaches that "[c]omputer programmers typically develop computer programs for conventional processors using relatively high-level source code computer languages (e.g., C++, Pascal, Fortran, etc.)." (Para. [0009]).

2. Pieper teaches that "During this compilation process, the prefetch instructions may be explicitly inserted into an intermediate level, machine-independent code that is first generated by the process from the input source code. (Para. [0020]).

3. Pieper teaches that "[a] later machine code-generation process may then translate/convert the intermediate level code, including the prefetch instructions, into machine-specific program instructions that are intended to be executed by the target processor." (Para. [0020]).

THE SECONDARY CAIN REFERENCE

4. Cain teaches "The BLAS library has been written so that it is *portable* to several platforms. It has also been *optimized* to the extent possible in *high-level* source code." (Page 1, ¶ 2, ll. 6-8, emphasis added).

ANALYSIS

We decide the question of whether Appellants have shown the Examiner erred in finding that the combination of Pieper and Cain teaches and/or suggests the argued limitation of "a resulting first optimized form of the software program is completely independent of the target processor." (*See* independent claims 1 and 14).

In the principal Brief, Appellants point to page 6, lines 7-8 of the Specification as providing support for the argued "first optimized form of the software program [that] is completely independent of the target processor" limitation. (App. Br. 3).

Lines 7-8 of page 6 of Appellants' Specification disclose: "Preferably, the programming language is one that is completely portable between all

probable DSP targets." To put this statement in context, for convenience we reproduce lines 3-11 of page 6 of Appellants' Specification, as follows:

> FIG. 1 is a flow diagram generally depicting steps in a process of optimizing a software program for a target processor. In the embodiment represented in FIG. 1, an optimization process 100 comprises three optimization steps. In a first optimization step 102, the software for the DSP processor target is written in a high-level language such as C, C++ or Ada. Preferably, the programming language is one that is completely portable between all probable DSP targets. In coding the software, optimization techniques particular to the language are preferably used. The code optimization in this step 102 preferably does not employ any optimization tools that depend on the processor that is meant to host the application.

(Spec. 6, ll. 3-11).

We particularly note that the aforementioned portion of the Specification discloses a first optimization step 102 where software for the target processor is written in a high level language where "[i]n coding the software, optimization techniques particular to the language are preferably used." (Spec. ll. 8-9). Thus, as supported by Appellants' Specification, the claimed "first optimized form of the software" is created by writing code in a high-level language using target independent optimization techniques particular to the language.

However, the Examiner counters that "[o]ne skilled in the pertinent art would know that a high-level language compiler forms intermediate code which is machine independent and optimizes the intermediate code in the middle end of the compilation stage, while formulating the target object

7

code, which is also optimized by using machine-specific optimizations at a later stage of the compilation. (Ans. 11, ¶2, ll. 4-8).

We find the Examiner's statement is supported by Pieper's teaching of "machine-independent code that is first generated by the process from the input source code . . . ." (FF. 2). Pieper also teaches the use of high-level languages (FF 1). Moreover, Pieper teaches that "[a] later machine code-generation process may then translate/convert the intermediate level code, including the prefetch instructions, into machine-specific program instructions that are intended to be executed by the target processor." (FF 3).

However, from our review of Pieper, we agree with Appellants that the *machine independent* code in Pieper is not the output of optimizer processes 58 that is clearly shown in Figure 2 as receiving an input from target machine model 80. Thus, the output of Pieper's optimizer 58 is code optimized for a particular target machine. We also agree with Appellants that intermediate program code that is *substantially independent* of the target processor architecture is not *completely independent* of the target processor (*see* Pieper, ¶ [0030]).

Nevertheless, we examine the totality of the evidence before us for indicia of obviousness.[2] In particular, we find the secondary Cain reference

---

[2] "Invention or discovery is the requirement which constitutes the foundation of the right to obtain a patent . . . unless more ingenuity and skill were required in making or applying the said improvement than are possessed by an ordinary mechanic acquainted with the business, there is an absence of that degree of skill and ingenuity which constitute the essential

clearly teaches a library of *portable* software programs that have been optimized in high-level source code: "The BLAS library has been written so that it is *portable* to several platforms. It has also been *optimized* to the extent possible in *high-level* source code." (FF 4).

Appellants have pointed to the disclosure in the Specification of a completely *portable* programming language (Spec. 6, lines 7-8) as providing support for the argued *first optimized form of the software program that is completely independent of the target processor*. (App. Br. 3). Thus, when we apply a broad but reasonable construction fully consistent with Appellants' Specification, we find Cain's portable, optimized high-level source code clearly teaches the argued limitations of "a resulting first optimized form of the software program [that] is completely independent of the target processor and is at least partially coded in the high-level language." (Claim 1).

While our reading of Appellants' claim 1 on the references has departed from some aspects of the Examiner's reading, it is our view that Appellants have been given full and fair notice of the references. Appellants are responsible for all that is disclosed by the collective teachings of the cited references. *See In re Zenitz*, 333 F.2d 924, 926 (CCPA 1964) ("This court has held in a number of decisions that a United States patent speaks for

---

elements of every invention." *Dunbar v. Myers*, 94 U.S. 187, 197 (1876) (citing *Hotchkiss* v. *Greenwood*, 52 U.S. 248, 267 (1850)). We note that *Hotchkiss* v. *Greenwood* was cited with approval by the Supreme Court in *KSR*, 127 S. Ct. at 1734, 1739, 1746.

all it discloses as of its filing date, even when used in combination with other references."). Our reviewing court has stated: "[t]he use of patents as references is not limited to what the patentees describe as their own inventions or to the problems with which they are concerned. They are part of the literature of the art, relevant for all they contain." *In re Heck*, 699 F.2d 1331, 1333 (Fed. Cir. 1983) (quoting *In re Lemelson*, 397 F.2d 1006, 1009 (CCPA 1968)). It is our view that this reasoning is equally applicable to non patent references, such as the secondary Cain reference we have addressed *supra*.

Even though, in some instances, we sustain rejections for different reasons than those advanced by the Examiner, our position is still based upon the collective teachings of the references and does not constitute a new ground of rejection. *See In re Bush*, 296 F.2d 491, 496 (CCPA 1961); *In re Boyer*, 363 F.2d 455, 458 n.2 (CCPA 1966).

For at least the aforementioned reasons, it is our view that the Examiner has set forth a sufficient initial showing of obviousness, and we find that Appellants have not shown error in the Examiner's rejection of representative claim 1 as being unpatentable over the combination of Pieper and Cain. Accordingly, we sustain the Examiner's rejection of representative claim 1, and of claims 2-26 which fall therewith.

CONCLUSION

Based on the findings of facts and analysis above, we conclude that Appellants has not established that the Examiner erred in rejecting claims 1-26 under 35 U.S.C. § 103(a) for obviousness.

DECISION

We affirm the Examiner's decision rejecting claims 1-26.

No time period for taking any subsequent action in connection with this appeal may be extended under 37 C.F.R. § 1.136(a)(1)(iv).

AFFIRMED

pgc

VISTA IP LAW GROUP LLP
1885 Lundy Avenue
Suite 108
SAN JOSE, CA 95131